Extraction de caractéristiques par codage neuro-prédictif

M. Chetouani, B. Gas, J.L. Zarader, C. Chavy

Laboratoire des Instruments et Systèmes d'Ile-De-France
Université Paris VI
BP 164 Tour 22-12 2éme étage
4 Place Jussieu, 75252 Paris Cedex 05
Mohamed.Chetouani@lis.jussieu.fr gas@ccr.jusieu.fr zarader@ccr.jussieu.fr chavy@ccr.jussieu.fr

ABSTRACT

In this paper, we present a predictive neural network called Neural Predictive Coding (NPC). This model is used for non linear discriminant features extraction (DFE) applied to phoneme recognition. We also, present an extension of the NPC model : NPC-3. In order to evaluate the performances of the NPC-3 model, we carried out a study of Darpa-Timit phonemes (in particular /b/, /d/, /g/ and /p/, /t/, /q/ phonemes) recognition. Comparisons with traditional coding methods are presented (LPC, MFCC and PLP) : they put in obviousness an improvement of the classification.

1. INTRODUCTION

Au cours des dernières années, de nombreux efforts de recherche ont été effectués afin d'améliorer les performances des systèmes de reconnaissance de la parole. La phase de codage joue un rôle très important dans la chaîne de reconnaissance. En effet, cette étape permet d'extraire des caractéristiques qui seront ensuite utilisées par le classifieur. La phase d'extraction de caractéristiques doit être faite avec soin, car elle contribue directement aux performances du système global. Les codeurs les plus couramment utilisés sont le codage linéaire prédictif (Linear Predictive Coding LPC), le codage cepstral (Mel Frequency Cepstre Coding MFCC) ou bien le codage linéaire prédictif perceptuel (Perceptual Linear Predictive PLP) [Her90]. Le codage MFCC et le codage PLP ont la propriété d'intégrer des connaissances du modèle auditif humain. Ces méthodes de codage sont mal adaptées pour traiter les non linéarités contenues dans les signaux de parole. Actuellement les approches utilisées sont multiples. On peut citer :

- les techniques à base d'analyse temps - fréquence ou d'analyse fréquentielle [Lon96], [Pal98].
- les analyses multi-résolution [Nit98].
- les modélisations du modèle perceptif humain [Kas97], [Mor00].
- les analyses factorielles : analyse discriminante [Sao00] ou en composantes principales [Lee00].

Dans cet article, nous présentons le codage neuronal prédictif (Neural Predictive Coding NPC) [Gas01a] qui est une extension non linéaire du codage LPC. La modélisation non linéaire permet d'intégrer des caractéristiques non linéaires des signaux de parole [Tea90], [Tow91], [Thy94]. Les structures des prédicteurs non linéaires sont essentiellement basées sur deux méthodes : le filtrage de Volterra [Che90] et les réseaux de neurones [Hus00]. L'avantage des filtres de Volterra est que les coefficients solutions de l'erreur quadratique minimale peuvent être exprimés de manière analytique. Par contre un défaut majeur des deux méthodes réside dans le nombre élevé de coefficients de codage.
Le codeur NPC a l'avantage de réduire ce nombre tout en conservant une modélisation non linéaire. Le modèle NPC a déjà été étendu par l'intégration d'informations de classes dès la phase de paramétrisation du modèle : le modèle NPC-2 [Gas01a], [Gas01b]. Cette méthode permet ainsi d'améliorer l'étape de classification. Nous proposons également une méthode de discrimination pour le modèle NPC : le modèle NPC-3. Cette méthode consiste à maximiser une mesure de discrimination : le *rapport d'erreur de modélisation*.
L'article est organisé de la manière suivante. Dans les sections 2 et3, nous décrivons brièvement les modèles NPC-1 et 2. Dans la section suivante, nous introduisons le *rapport d'erreur de modélisation* (MER) qui à la base du modèle NPC-3. Enfin, dans la section 5, nous comparons les modèles NPC avec les méthodes de codage traditionnelles dans une tâche de reconnaissance de phonèmes .

2. LE MODÈLE NPC-1

Le modèle neuro-prédictif (NPC) est une extension du modèle linéaire prédictif (LPC) pour la modélisation non linéaire des signaux de parole. Il est basé sur un réseau de neurones à une couche caché. La cellule de sortie est la cellule de prédiction (figure 1)

Les signaux de parole sont divisés en un nombre fixe de fenêtres afin de former des phonèmes de taille identique. La tâche de prédiction du modèle NPC consiste à prédire l'échantillon courant à l'aide des échantillons passés du phonème.

Si L est la taille de la fenêtre de prédiction, alors on a :

$$\hat{y}_k = \mathrm{F}(\mathbf{y}_k) \text{ avec } \mathbf{y}_k = \left[y_{k-1}, y_{k-2}, \cdots y_{k-L} \right]^{\mathrm{T}} \quad (1)$$

F est une fonction non linéaire qui est composée d'une fonction $\mathrm{G}_{\mathbf{w}}$ (associée à la couche cachée) et $\mathrm{H}_{\mathbf{a}}$ (associée à la couche de sortie)

$$\mathrm{F}_{\mathbf{w},\mathbf{a}} = \mathrm{H}_{\mathbf{a}} \circ \mathrm{G}_{\mathbf{w}} \quad (2)$$

$$\text{avec } \hat{\mathrm{y}}_k = \mathrm{H}_{\mathbf{a}}(\mathrm{z}_k) \text{ et } \mathrm{z}_k = \mathrm{G}_{\mathbf{w}}(\mathbf{y}_k)$$

w représente les poids de la couche cachée et **a** les poids de la seconde couche. Les poids sont déterminés par la minimisation de l'erreur de prédiction :

$$\mathrm{L} = \sum_k (y_k - \hat{y}_k)^2 = \sum_k \left(y_k - \mathrm{F}_{\mathbf{w},\mathbf{a}}(\mathbf{y}_k) \right)^2 \quad (3)$$

Où k est l'indice des échantillons.

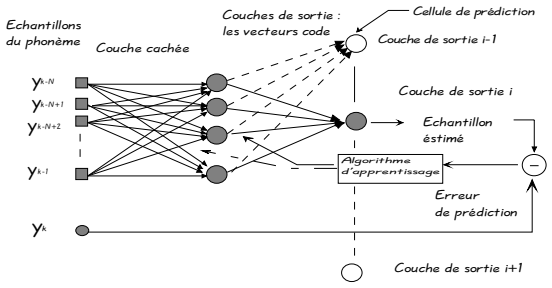

Figure 1 : Architecture du codeur NPC

L'idée clé du codeur NPC-1 est de limiter le nombre des coefficients de codage, en créant une seconde couche par phonème alors que la première couche est commune à tous les phonèmes. La fonction coût précédemment définie (équation 3) devient alors :

$$\mathrm{L} = \sum_i \sum_k \sum_l \left(y_{i,k} - \mathrm{F}_{\mathbf{w},\mathbf{a}_l}(\mathbf{y}_{i,k}) \right) \delta_{i-l} \quad (4)$$

Où $\mathrm{F}_{\mathbf{w},\mathbf{a}_l}$ est une des *M* fonctions associées aux $\mathbf{a}_l$ couches de sortie. δ est le symbole de Kronecker qui associe le phonème i à la couche de sortie *l*.

La phase d'apprentissage du codeur NPC s'effectue en deux étapes :

- La phase de paramétrisation. Cette phase consiste à déterminer tous les poids du réseau en utilisant les exemples des M classes de phonèmes. Les différentes secondes couches crées durant cette phase ne seront plus utilisées. Les poids de la première couche constituent les paramètres du codeur.
- La phase de codage. A la suite de la phase de paramétrisation, on présente un nouveau phonème destiné à être codé. Par minimisation de l'erreur de prédiction à l'aide des poids de la première couche, on détermine les poids de la seconde couche. Ces derniers forment le vecteur code du phonème.

3. LE MODÈLE NPC-2

Le modèle NPC-2 est proche du modèle NPC-1. La différence réside dans le fait que durant la phase de paramétrisation du modèle NPC-2, on introduit une notion de classes : une second couche est associée à une classe de phonèmes (et non à une occurrence de phonèmes comme c'est le cas pour le modèle NPC-1). La fonction coût devient alors :

$$\mathrm{L} = \sum_i \sum_k \sum_l \left(y_{i,k} - \mathrm{F}_{\mathbf{w},\mathbf{a}_l}(\mathbf{y}_{i,k}) \right) \delta_{C_i - l} \quad (5)$$

C_i représente la classe du phonème *i* parmi les *M* classes possibles. δ est le symbole de Kronecker qui associe la classe C_i à la couche de sortie *l*.

4. LE MODÈLE NPC-3

Pour garantir une extraction de caractéristiques discriminantes, on introduit une contrainte sur l'évolution des poids durant la phase de paramétrisation. Une des méthodes possibles consiste à introduire une discrimination explicite entre les modèles. Lorsque l'on effectue la paramétrisation d'un codeur NPC-2 par la classe du phonème i, on obtient le modèle NPC-2 suivant : $\mathrm{F}_{\mathbf{w},\mathbf{a}_i}$. Une mesure de discrimination entre les classes i et j consiste à déterminer l'erreur de prédiction du phonème i par le modèle NPC-2 $\mathrm{F}_{\mathbf{w},\mathbf{a}_j}$:

$$L_j^i = \sum_k \left(y_{i,k} - \mathrm{F}_{\mathbf{w},\mathbf{a}_j}(\mathbf{y}_{i,k}) \right)^2 \quad (6)$$

Le *rapport d'erreur de modélisation* Γ_{NPC} (Modelisation Error Ratio) [Gas01b] consiste à étendre ce calcul d'erreur de prédiction à l'ensemble des M classes :

$$\Gamma_{NPC} = \frac{Q^d}{(M-1)Q^m} \quad (7)$$

avec $Q^d = \sum_{i=1}^{M} \sum_{j=1, j\neq i}^{M} \mathrm{L}_j^i$ et $Q^m = \sum_{i=1}^{M} \mathrm{L}_i^i$.

L'objectif de l'apprentissage est de minimiser l'inverse du MER : $Q_{NPC3} = \frac{1}{\Gamma_{NPC}}$. (8)

D'après cette optimisation, la loi d'évolution des poids de la couche cachée ou de la seconde couche est proportionnelle à l'inverse du gradient du MER :

$$\frac{\partial}{\partial a}\left(\frac{1}{\Gamma_{NPC}}\right)=\frac{M-1}{Q^d}\left(\frac{\partial Q^m}{\partial a}-\frac{1}{\Gamma}\frac{\partial Q^d}{\partial a}\right) \quad (9)$$

Cette loi d'évolution fait apparaître deux termes :

- Le premier correspond à la minimisation de l'erreur de prédiction du codeur NPC-2
- Le second correspond à la maximisation de la mesure de discrimination (MER).

5. EXPERIMENTATION

Afin d'évaluer les performances des modèles NPC, nous les testons sur une tâche de reconnaissance de phonème.

5.1 Base de données

Nous extrayons différentes bases de la base Darpa-TIMIT [Tim]. L'évaluation se fait sur 3 bases. La première base est constituée de quatre classes de phonèmes voisés (les voyelles) : /aa/, /ae/, /ey/ et /ow/. Les deux autres bases sont formées d'occlusives : /b/-/d/-/g/ (voisés) et /p/-/t/-/q/ (non voisés). Les bases formées d'occlusives sont intéressantes car leur identification est réputée pour être difficile. Elles ont été utilisées par Lang et Waibel [Lan90] pour valider le modèle TDNN (Time Delay Neural Network). Les phonèmes sont choisis de manière aléatoire dans l'ensemble des locuteurs disponibles afin de produire un environnement multi-locuteurs. Chaque phonème est divisé, selon sa durée, en un nombre de fenêtres fixé à 256 échantillons.

5.2 Classification par Perceptron Multicouches

Afin de comparer les performances d'extraction de caractéristiques des modèles NPC, nous proposons de les comparer avec les méthodes de codage traditionnelles : LPC, MFCC et PLP. Le nombre de coefficients de codage est fixé à12. Nous utilisons un classifieur du type perceptron multicouches. Le classifieur possède 12 entrées (nombre de coefficients de codage), 10 cellules sur la couche cachée et autant de sorties que de classes de phonèmes.

5.3 Evaluation du modèle NPC

Dans ce paragraphe, nous allons présenter les résultats en classification des différentes bases de phonèmes en utilisant les différentes méthodes de codage : NPC-1, NPC-2 et NPC-3 ainsi que les méthodes de codage traditionnelles : LPC, MFCC et PLP. Nous mesurons le score en généralisation.

La modélisation non linéaire des phonèmes introduite par le modèle NPC permet d'extraire des informations utiles dans le cadre de la classification des voyelles (table 1). De plus, la modélisation des classes du modèle NPC-2 ainsi que la discrimination du codeur NPC-3 permettent d'augmenter les scores en classification.

Table 1: Score en généralisation pour les phonèmes : /aa/, /ae/, /ey/ et /ow/

LPC	MFCC	PLP	NPC-1	NPC-2	NPC-3
56.23%	58.25%	57%	61%	62.95%	65.25%

Les résultats sur la base /b/-/d/-/g/ sont concordants avec les résultats sur la base formée de voyelles, la base /b/-/d/-/g/ sont des phonèmes voisés (table 2). Dans cette table, on retrouve aussi les résultats pour les phonèmes non voisés : /p/-/t/-/q/. Les méthodes spectrales comme le codage MFCC ont de meilleures performances que les méthodes purement prédictives comme le codage LPC ou NPC. Cependant, la discrimination apportée par le modèle NPC-3 permet de surpasser cet handicap. Nous obtenons donc une meilleure extraction de caractéristiques même dans le cas de phonèmes non voisés.

Table 2 : Score en généralisation pour les occlusives

	LPC	MFCC	PLP	NPC-1	NPC-2	NPC-3
/b/-/d/-/g/	57.28%	62%	64%	65%	70.4%	73%
/p/-/t/-/q/	62.3%	66.6%	66%	63.33%	65%	70.3%

L'étude du rapport d'erreur de modélisation (MER) ainsi que celle de la variance inter-classes permettent de confirmer ces résultats (figure 2).

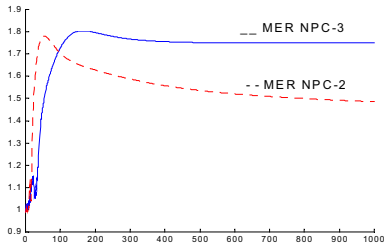

Figure 2: Rapport d'erreur de modélisation pendant la phase de paramétrisation pour la base formée de voyelles

Nous pouvons remarquer que le *rapport d'erreur de modélisation* (MER) constitue une mesure de discrimination entre les classes (figure 2). En effet, plus ce rapport est grand, plus la discrimination entre les classes est importante. Ce résultat se retrouve dans l'étude de la variance inter-classes (figure 3). Le codeur NPC-3 permet d'obtenir de meilleures

performances que le codeur NPC-2, car il maximise le rapport d'erreur de modélisation (MER), et de ce fait maximise la variance inter-classes.

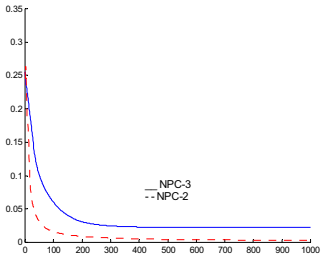

Figure 3: Variance inter-classes pendant la phase de paramétrisation pour la base formée de voyelles

6. CONCLUSIONS

Dans cet article, nous avons présenté une méthode d'extraction de caractéristiques : le codage neuro-prédictif (NPC). La modélisation non linéaire apportée par ce modèle est adaptée au traitement des signaux de parole. De plus, le codeur NPC-3 permet une extraction de caractéristiques discriminantes par le biais de la maximisation du *rapport d'erreur de modélisation* (MER). Nous avons aussi montré comment ce rapport permet de mesurer la discrimination entre les classes et cela pendant la phase de paramétrisation même. Grâce aux différentes caractéristiques du modèle NPC-3 (modélisation non linéaire et discrimination), les performances en classification sont meilleures que les techniques de codage traditionnelles. Ces résultats sont aussi vrais dans le cas de phonèmes non voisés. Nos futurs axes de recherche consistent à étendre l'étude du modèle NPC à l'ensemble des phonèmes. Pour cela, il est nécessaire d'optimiser les phases de paramétrisation et de codage, à l'aide du MER, pour extraire les caractéristiques les plus discriminantes possibles.

BIBLIOGRAPHIE

[Che90] P. Chevalier, P. Duvaut, B. Pincinbone (1990), « Le Filtrage de Volterra Réel et Complexe en Traitement du Signal », Traitement du Signal, Vol. 7, No. 5, pp. 451-476.

[Gas01a] B. Gas, J.L. Zarader, C. Chavy (2001), « A New Approach to Speech Coding : The Neural Predictive Coding », Journal of Advanced Computational Intelligence, Vol. 4, pp.120-127.

[Gas01b] B. Gas, J.L. Zarader, C. Chavy, M. Chetouani (2001), « Discriminant Features Extraction by Predictive Neural Networks», SSIP'01, pp.64-68.

[Her90] H. Hermansky (1990), « Perceptual linear predictive (PLP) analysis of speech », The Journal of the Acoustical Society America, pp. 1738-1752.

[Hus00] A. Hussain (2000), « Locally-Recurrent Neural Networks for Real Time Adaptative Nonlinear Prédiction for Non-Stationary Signals », Control and Intelligent Systems, Vol. 28, pp. 64-7.

[Kas97] K. Kasper, H. Reininger, D. Wolf (1997), « Exploiting the potential auditory preprocessing for robust speech recognition by locally recurrent neural networks », ICASSP, Vol. 2, pp. 1223-1226.

[Lan90] K.J. Lang, A.H. Waibel, G.E. Hinton (1990), « A Time-Delay Neural Network Architecture for Isolated Word Recognition », Neural Networks, Vol. 3, pp. 23-43.

[Lee00] J.H. Lee, H.Y. Jung, T.W. Lee, S.Y. Lee (2000), « Speech feature extraction using Independent Component Analysis », ICASSP, Vol. 3, pp. 1631-1634.

[Lon96] C.J. Long, S. Datta (1996), « Wavelet Based Feature Extraction for Phonème Recognition », ICLSP, Vol. 1, pp. 264-268.

[Mor00] R. De Mori, D. Albesano, R. Gemello, F. Mano (2000), « Ear-Model derived features for automatic recognition », ICASSP, Vol. 3, pp. 1603-1606.

[Nit96] T. Nitta (1998), « A Novel Feature-Extraction for Speech Recognition Based on Multiple Acousit-Features Planes », ICASSP, Vol. 1, pp. 29-32.

[Pal98] K.K. Paliwal (1998), « Spectral Subband Centroid Features for Speech Recognition », ICASSP, Vol. 2, pp. 617-620.

[Sao00] G. Saon, M. Padmanabban, R. Gopinath, S. Chen (2000), « Maximum Likelihood Discriminant Feature Spaces », ICASSP, Vol. 2, pp. 1229-1232.

[Tea90] H. Teager, S. Teager (1990), « Evidence for nonlinear sound production mechanisms in the vocal tract », Proc. NATO ADI on Speech Production and Speech Modeling, pp. 241-261.

[Tow91] B. Towshend (1991), « Nonlinear Prédiction of Speech », ICASSP, Vol. 1, pp. 425-428.

[Thy94] J. Thyssen, H. Nielsen, S.D. Hansen (1994), « Non-linearities short-term prédiction in speech coding », ICASSP, Vol. 1, pp. 185-188.

[TIM] University of Pennsylvania Linguistic Data Consortium. Darpa-Timit : a multi-speaker data base.